



Filtering High Quality Text for Display
on Raster Scan Devices

J. Kajiya
and
M. Ullner

Computer Science Department
California Institute of Technology

5018:TR:82

FILTERING HIGH QUALITY TEXT FOR DISPLAY
ON RASTER SCAN DEVICES

by J. Kajiya and M. Ullner
Computer Science Department
California Institute of Technology

Abstract. Recently several investigators have studied the problem of displaying text characters on grey level raster scan displays. Despite arguments suggesting that grey level displays are equivalent to very high resolution bitmaps, the performance of grey level displays has been disappointing. This paper will show that much of the problem can be traced to inappropriate antialiasing procedures. Instead of the classical $(\sin x)/x$ filter, the situation calls for a filter with characteristics matched both to the nature of display on CRTs and to the human visual system. We give examples to illustrate the problems of the existing methods and the advantages of the new methods. Although the techniques are described in terms of text, the results have application to the general antialiasing problem--at least in theory if not in practice.

1. INTRODUCTION

The computer age nearly destroyed quality printed and displayed text. Many of us remember our first sight of the ugly, uneven impression of a high speed chain printer. At the time, it seemed inevitable that high technology would sweep more beautiful--and less utilitarian--methods of text display aside for all but the most premium of uses. Recently this prospect has changed. With the growing availability of raster scan displays we have witnessed a technology with the capability of generating alphanumeric text that is more than just readable but pleasant to view as well.

It is an exciting dream of men like Donald Knuth to be able to compose locally and transmit for publication high quality text containing multiple fonts and mathematical equations. This dream would be made more attractive if an author would be able to see the result immediately, rather than having to wait several days for the output of a \$100,000 machine. The ideal would be to close the loop: to make available to the author an inexpensive real time device able to display high performance images.

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.

Furthermore, the effect of high quality real time displays on the activities of computer science itself has yet to be accurately assessed. But--aside from the creation of the word processing industry--the introduction of expanded codes to include relatively mundane features such as lower case characters has tremendously changed the flavor if not the substance of programming: surely none of us would wish to return to the 5-bit Baudot code! More importantly, the astonishing power and economy of computer languages and mathematical notations which incorporate special symbols (such as APL and symbolic logic) certainly has hidden lessons for the computer science community.

Many researchers soon discovered that the problem of displaying synthetically generated images on raster scan devices was a nontrivial task [Crow 1976, Blinn 1979]. The so called aliasing problem was encountered due to the high frequency content of artificially synthesized images. These researchers developed methods to overcome this problem which can be viewed alternatively as interpolation of brightnesses between pixels or filtering with a triangular convolution kernel. We shall, for definiteness, refer to this popular scheme as triangular filtering.

The first attempts to display text on raster scan devices used these intuitive filtering schemes that worked surprisingly well in practice [Warnock 1980, Seitz]. It is the aim of this paper to analyze the performance of these schemes for the general image case as well as the text case. We also propose a new method for choosing the pixel values which make up a synthetic image, show some preliminary results, and finally discuss the future directions that this research may take.

By now, the aliasing problem for computer generated images is well known to all in the field, as are the frequency domain interpretation of the phenomenon and the first order approximation to its solution. We wish to examine in detail the performance of this first order approximation. It is well known that the triangular filtering algorithm is cheap, fast, easy to implement, and produces an adequate antialiased image for very many applications. There are other applications, however, that require higher performance. Text is just such an application. Characters consist almost entirely of sharp edges and contain small subpixel features, such as serifs. Also, the processing of text can be done offline and the result stored in permanent memory. The method we present here is expensive, slow, and relatively hard to implement, but it produces higher quality images than the usual triangular filtering scheme. It is not now suitable for real time, or near real time, raster display applications.

It should be emphasized that these methods and analyses are applicable--at least in theory--to the general problem of antialiasing arbitrary images. We have chosen to focus on the display of text rather than arbitrary graphic objects because certain computational advantages accrue due to the small size of characters. If the computational implementations of these methods currently were not economically impracticable we would be reporting on the general synthetic image display problem as well.

2. AN ANALYSIS OF POPULAR ANTIALIASING SCHEMES LINEAR FILTERING WITH A TRIANGULAR PSF.

This section will analyze how well the triangular filtering algorithm does in removing aliasing while not otherwise distorting the picture. The details of the development are necessarily mathematical, but we present the key ideas here for those who want an overview of the section. There are two primary sources of error in the triangular filtering scheme. (1) The triangle interpolation kernel is not an ideal low pass filter and passes frequencies that are beyond the Nyquist limit. Thus, it is subject to aliasing. (2) The interpolation kernel does not take into account the reconstruction kernel. That is, it ignores the fact that pixels on the CRT display are Gaussian spots. The Gaussian spots are not ideal lowpass filters either and given the usual focus setting the frequency response of the reconstruction is far from flat. How well does the usual scheme work? The answer depends, of course, on the nature of the images displayed, and the mathematics tells you how to calculate the answer for your image. The gist of the answer, though, is that for certain images like text, there is plenty of room for improvement.

The most popular scheme for antialiasing is to linearly filter the input signal with some sort of interpolation kernel [Schafer and Rabiner 1973, Oetken, et. al. 1975]. We focus on the case of artificial images such as text and computer generated calligraphic and halftone images for display on raster scan frame buffers. There is some controversy about the characteristics of the optimum interpolation kernel with respect to the amount of ringing, and whether negative lobes are desirable [Gabriel 1977]. Many workers have settled on a triangular interpolation kernel as a simple compromise that gives good results in practice and is easy to compute [Crow 1976, Warnock 1980].

In the interest of concreteness we shall restrict the ensuing analysis to the triangular interpolation case; however, the reader can readily discern that the arguments involved are quite general. Parts of this analysis are similar to those found in [Pratt 1978].

The triangular PSF is shown in Figure 1 and is given by the equation:

$$k(\xi) = \begin{cases} 1 - \left| \frac{\xi}{T} \right| & -T \leq \xi \leq T \\ 0 & \text{otherwise} \end{cases}$$

To perform the sampling and reconstruction we convolve the ideal image with the interpolation kernel, sample with raster pitch T , and reconstruct by convolving the sampled signal with the reconstruction kernel. As is well known, these steps are best visualized in the frequency domain.

The Fourier transform of $k(\xi)$, the triangular kernel, is $K(\omega)$ as shown in Figure 2. This is compared with the ideal Nyquist kernel of $\sin(\xi/T)/(\xi/T)$.

The analytic expression for $K(\omega)$ is

$$K(\omega) = \frac{\sin^2\left(\frac{T}{2}\omega\right)}{\left(\frac{\omega}{2}\right)^2}$$

This figure clearly shows a possible source of aliasing error allowed by this kernel. Namely for an input image $f(\xi)$ with Fourier transform $F(\omega)$ the root mean square aliasing energy is given by

$$(2.1) \quad E_{alias}^2 = \int_{-\infty}^{\infty} |F(\omega)K(\omega)|^2 d\omega - \int_{-\frac{\pi}{T}}^{\frac{\pi}{T}} |F(\omega)K(\omega)|^2 d\omega$$

Now if F has most of its energy concentrated in the low frequencies then the aliasing error energy is quite small since the two terms in Equation (2.1) are roughly the same. Unfortunately, most artificial images, and especially text, have a Fourier spectrum that resides almost exclusively in the high frequency portions of signal space. To get some idea of the energy error involved, let us take a "line source", viz. a line of delta functions. This situation is to be met, for example, in the very thin strokes of classic Roman capitals, and in the diagonal strokes of a capital A for the Bodoni typeface.

In this case $f(\xi)$ approaches a Dirac delta function whose Fourier transform, in turn, approaches a flat spectrum. Equation (2.1) then gives

$$E_{alias}^2 = \int_{-\infty}^{\infty} |K(\omega)|^2 d\omega - \int_{-\frac{\pi}{T}}^{\frac{\pi}{T}} |K(\omega)|^2 d\omega$$

or

$$= \int_{-\infty}^{\infty} \left| \frac{\sin^2\left(\frac{T}{2}\omega\right)}{\left(\frac{\omega}{2}\right)^2} \right| d\omega - \int_{-\frac{\pi}{T}}^{\frac{\pi}{T}} \left| \frac{\sin^2\left(\frac{T}{2}\omega\right)}{\left(\frac{\omega}{2}\right)^2} \right| d\omega$$

the relative aliasing energy is roughly

$$\frac{E_{alias}}{\int |f(\xi)|^2 d\xi} \quad (\text{in dB}).$$

To perform the sampling step in the frequency domain we merely replicate the signal at the sampling frequency. Assume we have already folded in all the aliasing energy so that our modified signal appears as in Figure 3. Sampling now replicates this modified signal to something shown in Figure 4. Now we can reconstruct the signal by passing it through a reconstruction filter. If the reconstruction kernel is the Nyquist kernel,

$$\frac{\sin\left(\frac{\omega}{T}\right)}{\left(\frac{\omega}{T}\right)}$$

then the signal is a low pass filtered version of the original. However, in this matter we are not free to exercise a choice for our reconstruction kernel, except for a very limited range. The reconstruction kernels available to us are fixed by the physics of the output devices at our disposal, whether they be electrostatic printer, COM devices, or CRT based displays. To take the most common example, it is well known that the spot luminance distribution for a CRT is Gaussian, the variance of which is set by the focusing. At the proper focus point, a flat-field raster just becomes smooth, this is given roughly at the point

$$\sigma^2 = (.66T)^2$$

where σ is the standard deviation of the Gaussian,

$$g(\xi) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{\xi^2}{2\sigma^2}}$$

Now, the Fourier transform of this reconstruction kernel is

$$G(\omega) = e^{-4\pi^4\omega^2\sigma}$$

Thus the final output signal looks more like Figure 5 instead of appearing as in the normal case. In this picture we can see two sources of error arising from the mismatch of triangular and Gaussian kernel: Imaging errors and equalization errors.

Imaging errors are produced by the leakage of spurious copies of the original signal. This is given by

$$E_{\text{image}}^2 = \int_{-\infty}^{-\frac{\pi}{T}} \left| G(\omega) \sum_{k=-\infty}^{\infty} s(\omega - \frac{2\pi}{T}k) \right|^2 d\omega + \int_{\frac{\pi}{T}}^{\infty} \left| G(\omega) \sum_{k=-\infty}^{\infty} s(\omega - \frac{2\pi}{T}k) \right|^2 d\omega$$

When the Gaussian spot is focused properly this error is quite small. (Otherwise a flat field wouldn't have appeared flat.)

By far the more serious error is caused by the mismatch between the triangle and Gaussian frequency responses. Let us for the moment ignore the effects of aliasing and imaging, say by attempting to display an already perfectly bandlimited signal on a properly focused display ($\sigma = T/66$). In this case the overall modulation transfer function, is given by

$$H(\omega) = \frac{\sin^2\left(\frac{T\omega}{2}\right)}{\left(\frac{\omega}{2}\right)^2} e^{-\pi^4\omega^2\sigma}$$

This curve is plotted on a dB scale in Figure 6. Note that for the higher frequencies the MTF is down by almost 10 dB! Clearly, this amount of attenuation is causing a significant amount of sharpness loss, particularly in the fine features of high quality fonts.

Thus, if one is constrained to use to the linear filtering approach, a high frequency preemphasis is clearly called for--at the expense of an increased aliasing error tradeoff.

OTHER KERNELS

Many ad hoc schemes besides linear filtering have been proposed. Many are equivalent to linear filtering with triangular or other kernels. These include proportional weighting of the area of a given pixel covered, trapezoidal decomposition, contour smoothing, and nearest neighbor schemes.

One may wonder if the above remarks apply to all interpolation kernels as well as the triangle. Furthermore, there are a wealth of possible nonlinear schemes that come to mind. One can imagine an Edison-type programme involving a massive amount of experiment in order to converge on the correct solution. There is, however, a non ad hoc approach that is closely related to the roots of the Whittaker-Shannon sampling theorem, from which the original frequency domain analysis is derived.

3. OPTIMUM LINEAR SAMPLING AND RECONSTRUCTION

Instead of choosing an arbitrary kernel and calculating its performance, in this section we present an approach that calculates the optimum linear antialiasing filter for a given output restoration kernel. It turns out that this method has a flaw which is corrected in the next section. The flaw is that images with negative outputs will be generated.

In a way, we may think of the image sampling and reconstruction procedure as an function approximation problem. We are given as basis functions the Gaussian spots on a CRT. The question we may pose then is: "What are the optimum weights to linearly combine the basis vectors for approximation of the ideal signal?"

In other words, we are free to vary the brightness of each pixel spot (which is a Gaussian distribution) in order to make the reconstructed signal as "close" to the original ideal image as possible. In a CRT, the reconstructed image is given by a weighted sum of Gaussian bumps:

$$\text{Output Image} = \sum_{i=-\infty}^{\infty} x_i g_i(\xi)$$

In this equation, x_i represents the pixel value, and $g_i(\xi)$ represents a Gaussian distribution centered at the i^{th} pixel. See Figure 7.

Each Gaussian is a shifted version of a canonical bump:

$$(2.2) \quad g_i(\xi) = g_0(\xi - iT)$$

Now the question is, how do we measure the closeness of two images? Namely, given two images how do we assign a non-negative real number which corresponds to the distance between them? The choice of such a distance metric is a nontrivial task--a choice on which the ultimate visual quality of the images is strongly influenced. We will discuss the choice of other image metrics based on the human visual system below, but for now we choose a particular metric which has many pleasing analytic (if not visual) properties, the mean square metric. The distance between two images f_1, f_2 is given by

$$\text{dist}^2 = \|f_1 - f_2\|_2^2 = \int_{-\infty}^{\infty} |f_1(\xi) - f_2(\xi)|^2 d\xi.$$

The sampling problem may now be stated thus:

Given an input image $f(\xi)$, what are the optimum pixel values x_i minimizing the error between the original image and the reconstructed image? That is, find the values $\dots, x_{-1}, x_0, x_1, \dots$ minimizing

$$J(\dots, x_{-1}, x_0, x_1, \dots) = \int_{-\infty}^{\infty} \left| f(\xi) - \sum_{i=-\infty}^{\infty} x_i g_i(\xi) \right|^2 d\xi.$$

Now, in practice, there are only a finite number of sample points to be determined, say x_0, x_1, \dots, x_{n-1} . In order to minimize this functional we take its gradient and set it to zero.

$$\nabla J(x_0, \dots, x_{n-1}) = \left(\frac{\partial J}{\partial x_k} \right)_{k=0, \dots, n-1} = 0.$$

This give rise to a system of equations

$$\frac{\partial J}{\partial x_k} = \frac{\partial}{\partial x_k} \int_{-\infty}^{\infty} |f(\xi) - \sum_{i=0}^{n-1} x_i g_i(\xi)|^2 d\xi = 0 \quad (k=0, \dots, n-1)$$

or

$$2 \int_{-\infty}^{\infty} (f(\xi) - \sum_{i=0}^{n-1} x_i g_i(\xi)) (-g_k(\xi)) d\xi = 0 \quad (k=0, \dots, n-1).$$

Doing a little algebra we obtain:

$$\sum_{i=0}^{n-1} x_i \int_{-\infty}^{\infty} g_i(\xi) g_k(\xi) d\xi = \int_{-\infty}^{\infty} f(\xi) g_k(\xi) d\xi.$$

Now we use the identity (2.2) to get

$$\sum_{i=0}^{n-1} x_i \int_{-\infty}^{\infty} g_0(\xi - iT) g_0(\xi - kT) d\xi = \int_{-\infty}^{\infty} f(\xi) g_0(\xi - kT) d\xi.$$

The quantities on both sides of this equation have names,

$$\int_{-\infty}^{\infty} g_0(\xi - iT) g_0(\xi - kT) d\xi = R_g((i-k)T)$$

where $R_g((i-k)T)$ is the autocorrelation function of the Gaussian spot evaluated at the point $(i-k)T$. The quantity on the left is simply $R_{fg}(kT)$ the cross correlation of f and g at kT .

Setting $R_g((i-k)T) = a_{ik}$ and $R_{fg}(kT) = b_k$ the above normal equations take the familiar form

$$Ax = b.$$

Where A is an $n \times n$ matrix called the Gram matrix and b is an n -dimensional column vector, where n is the number of pixels in the output. The optimum pixel values are then given by the solution of this system of equations.

It is well known that the autocorrelation matrix A is of the so called symmetric Toeplitz form, viz. it is constant along the major diagonals. There exist fast methods to invert such matrices [Levinson 1947, Trench 1964]. These stem from the fact that there are not really n^2 independent elements but rather n . Inversion with the Levinson-Trench scheme is $O(n^2)$ instead of the usual n^3 .

Note that in the 2-dimensional case the matrix is no longer Toeplitz but rather Block Toeplitz, with Toeplitz sub-blocks, thus enabling significant economies in the storage and computation of the solution vectors [Kajiya 1981]. These savings can be quite significant since for a picture n pixels square, the full Gram matrix requires n^4 elements and take time $O(n^6)$ time to solve, for $n=512$ the straightforward inversion scheme is well beyond the capabilities of even the largest of computers, while the Levinson-Trench recursion is quite practical.

It may seem that for text character fonts, much of this discussion is moot since characters are quite small, say 10×13 pixels. However, even for this size, the matrices have 16900 elements, and for a 30×30 pixel font the full autocorrelation matrix requires almost a million entries!

An important point concerns the reconstruction kernels. If they were not Gaussians as in a CRT but rather Nyquist kernels as in the ideal case, then the Sampling theorem obtains. The Gram autocorrelation matrix reduces to the identity due to the orthonormality of the Nyquist kernels, and the cross correlation step corresponds to a perfect lowpass filtering and sampling operation.

Thus, the operation may be very roughly interpreted as follows:

To reconstruct with a given waveform, first filter by that waveform (take the inner product) then solve the matrix problem with the Gram matrix of autocorrelations. If the matrix is large we may be able to ignore edge effects and consider the matrix simply as a convolution with the "Green's function" of the Gram operator, which serves as an equalizing filter to flatten the response of the initial filter.

Thus, the above process is a linear process and, we might add, one that is quite familiar in certain circles. It is, however, inadequate from several standpoints.

The two major inadequacies are, first, positivity constraints stemming from the physics of light and the physics of the display devices and, second, the inadequacy of the least square image metric as a suitable model for vision. In the next section we discuss the first of these shortcomings, while in a later section we treat the second.

4. THE POSITIVITY CONSTRAINT

In this section we analyze the cause of the negative lobes output by the optimum linear filter. We also explore methods for correcting the negative lobe output. It turns out that the obvious method of truncating the negative lobes at zero may or may not work, depending upon the form of the restoration kernel. We analyze the criteria under which truncation works. Unfortunately, for the case of interest, viz. Gaussian reconstruction kernels, the criteria are not met.

Figure 8 shows the minimum mean square error reconstruction of an impulse using Gaussian reconstruction kernels.

The relative extrema represent the strongest contributions of each individual Gaussian spot. This response can be couched in almost teleological terms as follows:

To make an impulse with a series of Gaussian bumps, take an initial bump and shave off the sides to narrow the bump by subtracting a small Gaussian from either side. Now to compensate for these negative lobes we add in some positive Gaussians of smaller proportions a little farther away. Now to compensate for these positive lobes,....etc.

This procedure cannot be followed if we have, say, a series of potentiometers controlling the brightness of a number of Gaussian spots. This is because the pots cannot be turned negative. There is no way to make negative light--much to the frustration of many workers concerned with these kind of display problems. Thus the display of a reconstructed image is constrained to the positive cone $x_0 > 0, x_1 > 0, x_2 > 0, \dots, x_n > 0$.

This puts an additional constraint on the sampling problem. A succinct statement of the problem is now:

Sampling problem (with positivity):

$$\text{Minimize } \int_{-\infty}^{\infty} |f(\xi) - \sum x_i g_i(\xi)|^2 d\xi$$

with x restricted to the positive cone

$$x_i \geq 0 \quad i = 0, 1, \dots, n-1.$$

How do we approach this problem? Well, one popular method has been to ignore it completely: simply solve for the unconstrained optimum reconstruction and set any negative values to zero. This method may work in certain cases. For example, if the picture is sufficiently bright everywhere, the negative lobe may never dip below zero. Whenever we need to truncate negative values, however, possibly severe inaccuracies result. Characters, lines, and many other graphic objects are binary pictures with the lesser value being zero. Thus in graphics the need to truncate arises often.

The typical case is illustrated by the previous example, viz. sampling an impulse. Simply truncating the negative lobes leaves a curious "ringing" pattern around the impulse (Figure 9). The ringing pattern in no way contributes to the minimization of the mean square metric, since their principal function was to compensate for the negative lobes. Setting the positive sidelobes to zero also happens to be very close to the minimum mean square error picture: a single Gaussian bump.

Geometrically, the situation may be visualized as in Figure 10. We have suppressed all dimensions except two and drawn contour lines for error. The actual optimum can be seen to be at point A in which $x_1 > 0$ but $x_2 < 0$. Setting $x_2 = 0$ projects the point A onto B, a point which satisfies the constraints but which isn't very close to the true constrained minimum given by point C.

How badly do we do by setting co-ordinates to zero? That depends on the eccentricity of the ellipse and the angles that the major axis of the ellipse forms with the co-ordinate axes. With a nearly round ellipse, one whose major and minor axes are very nearly equal in length, one comes very close indeed to the optimum, when one sets the offending co-ordinates to zero (Figure 11). In the case of reconstructing say a 10x13 pixel character we are confronted with a 130 dimensional ellipsoid. The ellipsoid is formed from the level surfaces of the mean square error functional. To find its eccentricity we merely find the ratio of the largest to the smallest of the eigenvalues of the Hessian of the (quadratic) error functional, i.e. the matrix of second order partial derivatives

$$\left(\frac{\partial^2 J}{\partial x_i \partial x_j} \right)$$

This is none other than our old friend the Gram autocorrelation matrix. Thus we find that the set of eigenvalues, and hence the eccentricity of the ellipsoid, depends on the reconstruction kernel.

Now, for an orthonormal reconstruction kernel, such as the Nyquist kernel, the Hessian is the identity matrix. Hence, the eigenvalues are all unity and the ellipsoid is perfectly round. Therefore, simply setting the offending co-ordinates to zero gives the constrained optimum exactly.

We do not have Nyquist kernels at our disposal, however, but rather Gaussians--which are decidedly not orthonormal. It turns out that finding the eigenvalues and eigenvectors of the Gram autocorrelation matrix is a well-known procedure called the Karhunen-Loeve transformation. Speaking very loosely, the eigenvalues correspond to the values of the Power spectral density, viz. the square of the Fourier transform magnitude of the autocorrelation function. But the autocorrelation of a Gaussian distribution is again a Gaussian with double the standard deviation, and it's well known that the Fourier transform of a Gaussian is also a Gaussian. Thus, there is a tremendous range in the magnitude of the eigenvalues encountered, the values being governed by an exponential of a term proportional to the square of the abscissa. In other words for a Gaussian distribution the ellipsoid is very eccentric. Furthermore, the angle of inclination of the ellipsoid with respect to the co-ordinate hyperplanes ($x_i = 0$) is given by direction cosines that correspond to the inner product of a Gaussian with the Karhunen-Loeve eigenvectors. These eigenvectors are very roughly sinusoids. These direction cosines, at least for the largest eigenvalues, are roughly equal, so the ellipsoid is pitched at an angle of about 45°--the worst case.

To sum up the discussion so far: While setting the negative coordinates to zero for orthonormal basis functions is a very good procedure, for the Gaussian restoration kernels it is very bad.

THE KUHN-TUCKER CONDITIONS

Since simply truncating the negative lobes of the output signal will not provide an optimum solution subject to the positivity constraint, we must search for methods that will provide us with the optimum constrained solution to the antialiasing problem. Note that we now are talking about some non-linear filtering procedure that will provide us with the optimum sample values for some input image. Fortunately, the structure of the aliasing problem is such that certain key conditions are met. This simplifies the optimization problem immensely and gives us a relatively straightforward way to solve the problem. These conditions are the Kuhn-Tucker conditions and the method of solution is known as the method of feasible directions.

To optimize a nonlinear functional subject to inequality constraints is, in general, a very difficult task. If the functional and its constraints satisfy the following assumptions then we may apply the Kuhn-Tucker theorems for inequality constrained mathematical programming problems. The Kuhn-Tucker conditions are:

- (i) The functional is convex. That is if x_i and x'_i ($i=0,1,\dots,n-1$) are two points in the solution space, then

$$J(\alpha x_i + (1-\alpha)x'_i) \leq \alpha J(x_i) + (1-\alpha)J(x'_i) \quad 0 \leq \alpha \leq 1.$$

In other words, the functional lies below a line joining any two of its values. See Figure 12.

- (ii) The feasible set, i.e. the set of points satisfying the constraints, is convex and has a nonempty interior.

Fortunately, both these assumptions hold for the case at hand: reconstruction with Gaussian kernels and positive weights. Its evident that the square error metric is convex (for other visual metrics this condition may no longer hold). The second assumption is also satisfied. The positive cone is obviously convex and has a generous interior.

We now state a Kuhn-Tucker theorem.

Kuhn-Tucker Theorem. Under the conditions mentioned above, the nonlinear programming problem (Equation 4.1) has a minimal solution $x^* > 0$ iff there exists $\lambda^* > 0$ such that the Lagrangian

$$J(x, \lambda) = J(x_0, \dots, x_{n-1}) - \sum_{k=0}^{n-1} \lambda_k x_k$$

has a saddle point at (x^*, λ^*) .

A useful interpretation of this theorem [Collatz and Wetterling 1975] is that at the optimum point x^* , the gradient of the functional is perpendicular to the active constraining hyperplanes, viz. the coordinate hyperplanes in which x^* has a zero component. If there is no constraining active hyperplane then the gradient must be zero.

The Kuhn-Tucker theorem provides the basis for a number of different optimization algorithms. One of the simplest (and slowest) is the one we have chosen in this work: the method of feasible directions. In this iterative method, a point x^i is updated by a vector proportional to the gradient of the functional projected upon a subspace which maintains the new iterate in the positive cone:

$$\text{Namely, } x^{i+1} = x^i - \epsilon_i \nabla J_P$$

where ∇J is the gradient of the functional.
P is the projection operator which

limits ∇J to a feasible subspace, and
make $\epsilon_i > 0$ is a sequence of numbers chosen to
iteration the Jacobian decrease at each
(Steepest descent).

There are several salient points about this method which should be mentioned.

First, this method is nonlinear, e.g.

$$f \mapsto \hat{f} \quad \text{does not imply that} \quad c \cdot f \mapsto c \cdot \hat{f}$$

In particular, if c is chosen to make the bulk of $c \cdot f$ negative, then the output will be zero.

Second, the method can in certain cases collapse to the unconstrained case. For an input image that lies deep in the feasible set, i.e. it is positive everywhere, then one can afford the luxury of negative lobes because the ultimate answer will still have only positive coefficients. For an input image on the boundary of the feasible set, i.e. one that has many pixels set to zero, the method will suppress negative lobes. The next section will demonstrate how these constraints control ringing.

5. RESULTS

The above algorithm was programmed on a DECSYSTEM-20 for both the one and two dimensional cases. For the input images we hand digitized characters on either a 1x100 or 100x100 grid. The decimation ratio was set to 100:16 or 6.25. The coefficients were then reconstructed with artificial Gaussian distributions of known variances.

Results for the one dimensional case are as follows: An impulse response centered on a sample value gives the identical answer as the triangular kernel interpolant, a single Gaussian spot. Also shown is the unconstrained optimum for a box, which appears in Figure 13 as a ringing sinc-like function.

Figure 14 shows the effect of constraints upon the negative lobes of a step response. Note that suppression occurs for not only the negative sidelobes but also for the residual positive sidelobes. We still have ringing on the positive portions of the step, however. These can be removed by a range constraint: if we know that the images have values between 0 and 1 (as do many graphic objects) then constraining the x_i to $0 < x_i < 1$, gives the reconstruction shown in Figure 15. Finally the response of the algorithm to a chirp signal is shown for comparison with the triangular case.

For the two dimensional case, we reconstructed several characters with an artificial Gaussian spot. Rather than reconstruct with the natural electron beam spot on the CRT we chose to reconstruct with a much larger simulated spot. We did this for several reasons.

First, the pictures were easier to analyze for artifacts.

Second, at the time of writing we had as yet not measured the variance of the spot on the screen.

Third, the image display available to us had only 4 bits per pixel both for the memory and the colormap. Additionally, the gamma of the display system was not adequately compensated for (this was a display intended for VLSI design aids). It is well known that for anti-aliasing experiments proper gamma correction is crucial [Crow 1975]. Rather than lose precious bits by trying to gamma correct in the color map (remember we only have 16 levels in and out) we decided to reconstruct with large spots, gamma correct with high precision inside the computer and dither down to four bits, trading spatial resolution for gray scale resolution.

Figures 16-18 show the results of the algorithm compared with the original and the triangular filter reconstruction.

6. FUTURE WORK

We see many ways to continue this work.

We have not addressed at all the important problem of overlapping images. At present the expedient we use is to place each character in frames that does not overlap. If we were to allow overlapping frames then we would be allowing more pixels per character to yield a higher effective output resolution. There are dangers in allowing overlapping images, however. Because characters almost never occlude one another in the text case there is usually no danger when one is performing linear processing: one simply adds the resulting images. In the case of our nonlinear processing the situation is more delicate. A careful analysis has yet to be done.

The foremost task yet to be done is the inclusion of a better visual metric than the least squares metric. It will be clear from the following discussion that a minimum mean square reconstruction is quite far from optimum compared with a reconstruction which takes into account certain key features of the human visual system.

THE SINGLE CHANNEL MODEL

The simplest model of the visual system is the so-called Lateral Inhibition model (also known as the single channel model). The model is given by the following equation

$$(6.1) E_{\text{error}}^2 = \int_{-\infty}^{\infty} |h(\xi) * (\log f(\xi) - \log \hat{f}(\xi))|^2 d\xi$$

Where $f(\xi)$ is the input image, $\hat{f}(\xi)$ is the reconstructed image and $h(\xi)$ is a PSF known as the lateral inhibition kernel (or Ratliff kernel). Thus, in this model, each of the individual images to be compared undergoes a logarithmic point transformation after which the difference is filtered and then summed in a mean square procedure. The frequency response corresponding to $h(\xi)$ is shown in Figure 19 where the peak of the response curve is at about 3 cy/deg [Cornsweet 1971].

Let us analyze this image metric a bit. Recall that the Parseval theorem relates mean square error in the spatial domain to that in the frequency domain [Rudin 1966]:

$$\int_{-\infty}^{\infty} |f(\xi) - g(\xi)|^2 d\xi = \int_{-\infty}^{\infty} |F(\omega) - G(\omega)|^2 d\omega.$$

Applying this formula to the expression for the visual model response we obtain (via the convolution theorem)

$$E_{\text{error}}^2 = \int_{-\infty}^{\infty} |h(\xi) * (\log f_1(\xi) - \log f_2(\xi))|^2 d\xi = \int_{-\infty}^{\infty} |H(\omega)|^2 |\tilde{F}_1(\omega) - \tilde{F}_2(\omega)|^2 d\omega$$

where $H(\omega)$ is the Fourier transform of $h(\xi)$, $\tilde{F}_1(\omega)$ and $\tilde{F}_2(\omega)$ are the Fourier transforms of $\log f_1$ and $\log f_2$. Note that in this form the response is just a classical weighted least square error metric between the logs of the images. From the MTF of the visual system (Figure 19) we see that this weighting favors the high frequencies much more than the low frequencies.

There are two essential features that cause the optimum image reconstruction for the ordinary least square metric and for the lateral inhibition model metric to differ significantly. The first concerns the relative importance given to errors at different luminance levels and the second concerns the relative importance given to errors at different spatial frequencies.

The logarithmic point transformation, which forms the nonlinearity in the initial stage of the model, implements Weber's law:

$$k = \frac{\Delta I}{I}$$

where ΔI is the just noticeable difference in luminance and I is the average luminance. This law holds over a range of intensities that easily encompasses the range encountered in graphics and text display. It has been found that $k \approx 0.02$. The precise form for the function implementing Weber's law is in dispute. However, all proposed functions are reasonably close to the logarithm for a range of intensities and share most of its important

properties--such as concavity. In words, this law states that small luminance errors for the low luminance portions of an image are far more objectionable than those for the high luminance portions.

An optimal approximator using this image metric will thus be more fastidious about the low luminance portions of an image, spending more of its error budget to fit the data there. Since the human visual system is more sensitive to errors in this portion (hence the logarithm in the model) images reconstructed in this manner should compare favorably to those of the ordinary least squares metric in which all the errors are treated as equals.

A second difference is manifested by the frequency weighting which appears in Equation 6.1. Here the model correctly predicts that we are by far more sensitive to high spatial frequency errors. It is this increased sensitivity at high frequencies that makes the artifacts of fuzzy edges produced by the triangular kernel--and the persistent ringing produced by the unconstrained linear least squares approximant--so objectionable. Both artifacts are high frequency effects whose effect is enhanced by our visual system. Using the lateral inhibition metric will result in improved image sharpness and lower ringing at the expense of higher errors at the low frequency portions of the image, which presumably we do not see.

The methods for calculation of an optimal approximator for such an image metric have yet to be resolved. Introduction of the logarithmic nonlinearity (or any other nonlinearity popular in visual modelling) causes a loss of convexity for the functional. Thus, the Kuhn-Tucker theorems may not be applied directly. We are currently investigating implementations which will bypass this difficulty.

MULTICHANNEL MODELS

More realistic visual models can be considered for use in the optimal approximant scheme. Currently popular in the psychophysical literature are a class of models known as multichannel models [Graham and Nachmias 1971, Mostafavi and Sakrison 1976, Kajiya 1979]. In these models the image, after passing through a nonlinearity, is not filtered by a single frequency shaping network but rather by many bandpass channels of varying sensitivities. There is a certain amount of controversy over the characteristics of such channels and the mode of summation of the outputs of such channels, but it is a promising possibility that L metrics rather than L metrics may be closer to the truth. If this is the case then the door is open for nonlinear Chebyshev techniques to be applied to the antialiasing problem.

CONCLUSIONS

There is far more to the antialiasing problem than simple linear filtering. We have analyzed the performance of the linear filtering approach to antialiasing, and introduced the use of more powerful techniques for certain critical applications such as the display of high quality text. Perhaps someday computational techniques will be discovered to perform these calculations for more general synthetic images. For now, though, the practical use of our technique is limited to images with relatively small pixel sizes--such as the display of text characters.

REFERENCES

- Blinn J.F. (1978) "Computer Display of Curved surfaces" U. Utah Ph.D. Thesis, December 1978.
- Collatz L. and Wetterling W. (1975) OPTIMIZATION METHODS, Springer Verlag, Berlin.
- Cornsweet T. (1970) VISUAL PERCEPTION, Academic Press, New York.
- Crow F.C. (1976) "The aliasing problem in Computer-synthesized shaded images" U. Utah Computer Science Tech Rept. UTEC-Csc-76-015.
- Gabriel S.A. (1977) Private communication.
- Graham N. and Nachmias J. (1971) "Detection of grating patterns containing two spatial frequencies: a comparison of single channel and multiple channel models" Vision Res., v. 11, pp. 251-259.
- Kajiya, J.T. (1979) "Toward a Mathematical Theory of Perception", Ph.D. Thesis, U. Utah.
- Kajiya J.T. (1981) "On fast methods for two dimensional spectral factorization", to appear.
- Levinson N. (1947) "The Wiener RMS error criterion in filter design and prediction" J. Math. Phys. v.25, no. 4, pp.261-278.
- Mostafavi H. and Sakrison D.J. (1976) "Structure and properties of a single channel in the human visual system" Vision Res., v. 16, pp. 957-968.
- Oetken G., Parks T.W., Schuessler H.W. (1975) "New results in the design of digital interpolators" IEEE Trans. ASSP, v. ASSP-23, pp.301-309.
- Pratt W.K. (1978) DIGITAL IMAGE PROCESSING, Wiley-Interscience.
- Rudin W. (1966) REAL AND COMPLEX ANALYSIS, McGraw-Hill, New York.
- Sachs M.B., Nachmias J. and Robson J.G. (1971) "Spatial frequency channels in human vision" J. Optical Soc. Am., v. 61, pp. 1176-1186.
- Sakrison D.J. (1977) "On the role of the observer and a distortion measure in image transmission" IEEE Trans. on Communications, v. COM-25, pp. 1251-1267.
- Schafer R.W. and Rabiner L.R. (1973) "A digital signal processing approach to interpolation" Proc. IEEE v.61, pp.692-702.
- Trench W.F. (1964) "An algorithm for the inversion of finite Toeplitz matrices" J. SIAM v. 12, no. 3, pp.515-522.
- Seitz C., et. al. "Digital Video Display system with a Plurality of Grey-scale levels" US Patent 4,158,200.
- Warnock J.E. (1980) "The Display of Characters Using Gray level Sample arrays" ACM SIGGRAPH80, pp.302-307.

Figure 1.

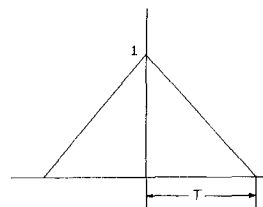


Figure 2.

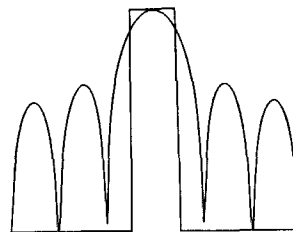


Figure 3.

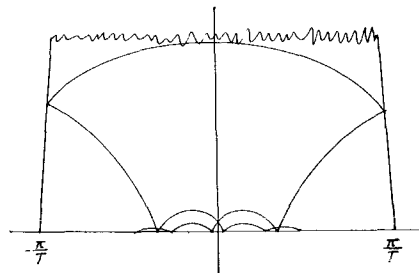


Figure 4.

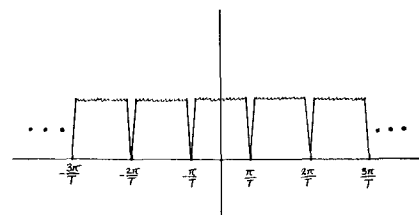


Figure 5.

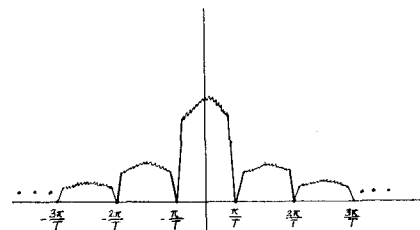


Figure 6.

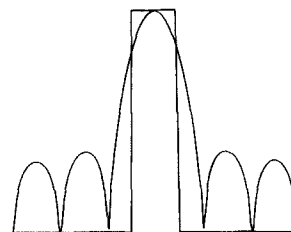
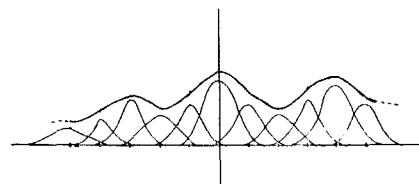


Figure 7.



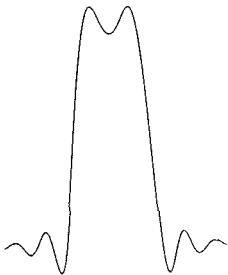


Figure 8.

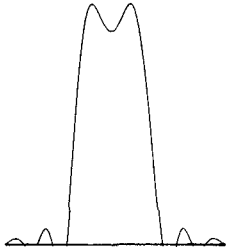


Figure 9.

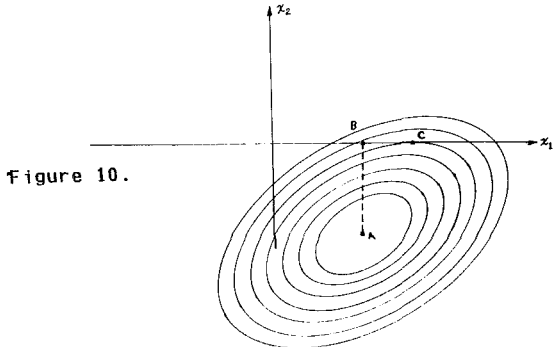


Figure 10.

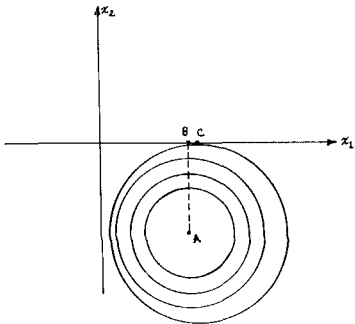


Figure 11.

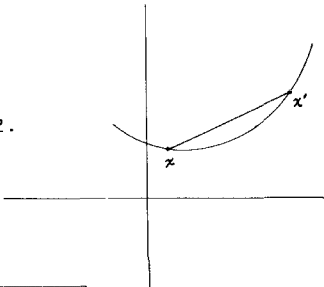


Figure 12.



Figure 16.



Figure 17.



Figure 18.

Figure 13.

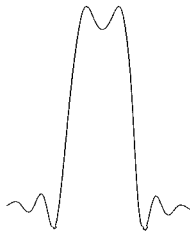


Figure 14.

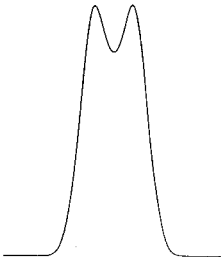


Figure 15.

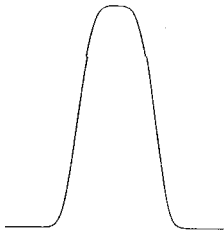


Figure 19.

